

**Listing of Claims**

This listing of claims will replace all prior versions, and listings, of claims in the application:

1-20 (Canceled)

21. (Previously presented) A method for providing an event system in a distributed directory-enabled application environment using an eXtensible Markup Language ("XML") application program interface, the interface including at least one event delineated by an event parameter, the event defining an object delineated by an object property and an object parameter, the method comprising:

defining at least one subscription filter allowing a subscriber to selectively filter the event;

publishing the event;

retrieving a subscriber list, the list including the subscriber and the subscription filter;

selecting the subscriber and the subscription filter from the list;

filtering the event through the subscription filter;

passing the event to the subscriber if the event passes through the subscription filter; and

acting on the event.

22. (Previously presented) The method of claim 21 further including filtering the event through an event filter based on the event parameter.

23. (Previously presented) The method of claim 22 further including acting on the event by altering the event parameter, so that the event is transformed through the alteration.

24. (Previously presented) The method of claim 21 further including filtering the event through an object filter, whereby the event is filtered based on the object property.

25. (Previously presented) The method of claim 24 further including acting on the event by altering the object property, so that the event is transformed through the alteration.

26. (Previously presented) The method of claim 21 further including filtering the event through a parameter filter, whereby the event is filtered based on the object parameter.

27. (Previously presented) The method of claim 26 further including acting on the event by altering the object parameter, so that the event is transformed through the alteration.

28. (Previously presented) The method of claim 21 further including acting on the event by terminating the event, so that the event is stopped by the termination.

29. (Previously presented) The method of claim 21 further including tracking the event, whereby information about the event may be gathered.

30. (Previously presented) The method of claim 29 further including generating reports from the gathered information, so that the information about the event may be compiled.

31. (Previously presented) A computer system for providing an event system in a distributed directory-enabled application environment using an eXtensible Markup Language ("XML") application program interface, the interface including at least one event delineated by an event parameter, the event defining an object delineated by an object property and an object parameter, the system comprising:

at least one processor;

at least one memory accessible to the processor;

a first application stored in a first portion of the memory;  
a second application stored in a second portion of the memory;  
defining at least one subscription filter allowing a subscriber to selectively filter the event;  
parsing the event with an XML parser;  
instantiating the event as a Data Object Model (DOM) class when services required by the event are unavailable;  
publishing the event by the first application;  
retrieving a subscriber list, the list including the subscriber and the filter;  
selecting the subscriber and the filter from the list;  
filtering the event through the filter;  
subscribing to the event by the second application if the event passes through the filter; and  
acting on the event by the second application.

32. (Previously presented) The computer system of claim 31 further including instructions for filtering the event through an event filter, whereby the event is filtered based on the event parameter.

33. (Previously presented) The computer system of claim 32 further including instructions for acting on the event by altering the event parameter, so that the event is transformed through the alteration.

34. (Previously presented) The computer system of claim 31 further including instructions for filtering the event through an object filter, whereby the event is filtered based on the object property.

35. (Previously presented) The computer system of claim 34 further including instructions for acting on the event by altering the object property, so that the event is transformed through the alteration.

36. (Previously presented) The computer system of claim 31 further including instructions for filtering the event through a parameter filter, whereby the event is filtered based on the object parameter.

37. (Previously presented) The computer system of claim 36 further including instructions for acting on the event by altering the object parameter, so that the event is transformed through the alteration.

38. (Previously presented) The computer system of claim 31 further including instructions for acting on the event by terminating the event, so that the event is stopped by the termination.

39. (Previously presented) The computer system of claim 31 further including instructions for tracking the event, whereby information about the event may be gathered.

40. (Previously presented) The computer system of claim 39 further including instructions for generating reports from the gathered information, so that the information about the event may be compiled.

41. (Previously presented) A method for providing an event system in a distributed directory-enabled application environment using an eXtensible Markup Language ("XML") application program interface, the interface including at least one event delineated by an event parameter, the event defining an object delineated by an object property and an object parameter, the method comprising:

- defining at least one subscription filter allowing at least first and second subscribers to selectively filter the event;

- publishing the event;

- retrieving a subscriber list, the list including the first and second subscribers and the subscription filter;

- selecting the first and second subscribers and the subscription filter from the list;

passing the event to the first subscriber if the event passes through a  
subscription filter;

modifying the event based on an alteration of at least one of the event parameter  
and the object parameter by the first subscriber; and

passing the modified event to the second subscriber.